

REMARKS/ARGUMENTS

This Amendment is in response to the Office Action mailed August 4, 2009.

Before this Amendment, claims 1-7, 12-13, 15-17, 44-52, and 54-56 were pending. In this Amendment, no claims have been amended, no claims are canceled, and no new claims are presented. Upon entry of this Amendment, which is respectfully requested, claims 1-7, 12-13, 15-17, 44-52, and 54-56 will be pending. Because no claims have been amended, this Amendment does not raise new issues requiring further search and/or consideration.

I. CLAIM REJECTIONS UNDER 35 U.S.C. § 103

The Office Action rejected claims 1-3, 7, 12-13, 15-17, 44-46, and 50-52 under 35 U.S.C. § 103(a) as being unpatentable over Walker et al. (US 2006/012281) (hereinafter “Walker”) in view of Chao et al. (US 2002/0169678) (hereinafter “Chao”). The rest of the claims (claims 4-6, 47-49, and 54-56) were rejected under 35 U.S.C. § 103(a) as being unpatentable over Walker in view of Chao and further in view of Chang et al. (US 2007/0226027) (hereinafter “Chang”). To establish a prima facie case of obviousness, the prior art reference, or references when combined, must teach or suggest all of the claim limitations. Applicants respectfully traverse the rejections because the cited references fail to teach or suggest all of the claim limitations.

For example, claim 1 recites in part:

storage configured to store generated allocation rules and to store transaction data associated with a plurality of transactions, each generated allocation rule being associated with at least one of the plurality of business objects, each generated allocation rule being **generated by combining a first predefined rule of a node of a hierarchical data structure with a second predefined rule inherited from a parent node**, the first predefined rule characterizing a member of the at least one of the plurality of business objects;

a query engine configured to query the transaction data using the generated allocation rules; and

an allocation manager configured to make one or more attempts to allocate a member of the plurality of transactions among the plurality of business objects,

wherein each generated allocation rule determines if a business object is entitled to an allocation from a particular transaction.

(emphasis added). Independent claim 44 recites similar limitations. Independent claims 12 and 51 recite some similar limitations.

Walker is directed to vending machines that dispense the usual convenience food products as well as “mystery packages” for promotional purposes (see, e.g., Walker abstract). The Office Action avers that Walker paragraph [0433] discloses each generated allocation rule being generated by combining a first predefined rule of a node of a hierarchical data structure with a second predefined rule inherited from a parent node. Applicants respectfully disagree.

Although Walker allocates products (including its mystery packages) available for sale based on stored rules, Walker’s allocation rules are not generated by combining one predefined rule with a second predefined rule **inherited** from a parent node as claimed. Furthermore, Walker’s inventory database that manages vended products does not appear to have nodes of a hierarchical data structure as claimed (see Walker paragraph [0434-0435] including Table 13). Walker’s soft drinks, candy, and other dispensed products are merely categorized by product “Type” (see third column from left in Table 13). A product type is not a node in Walker, it is a categorization. Even if ‘product type’ were erroneously considered a node, Walker’s various rules for dispensing products do not **inherit** rules from the product type. Thus, Walker does not disclose “each generated allocation rule being **generated by combining a first predefined rule of a node of a hierarchical data structure with a second predefined rule inherited from a parent node**” (emphasis added) as claimed.

Walker’s stored rules are for dispensing soft drinks and other junk food do not “determine[] if a business object is **entitled** to an allocation from a particular transaction” as claimed. Presumably, the owner of the vending machine allocates no commission to the brand owners of the products because the owner has purchased the low cost candy, soda, and other items outright and keeps the profits to himself. For example, the vending machine owner keeps a margin of \$.40 for each Coca-Cola® drink sold (see Walker Table 14 in paragraph [0441]). There is no incentive to pay a salesperson because a vending machine sells its inventory itself.

Walker does not teach or suggest the above limitations. Walker's product allocation rules are based on individual sales rates and value scores of its various products, not inheritance. Walker's rules do not determine whether a business object is entitled to an allocation. There is no salesperson to compensate because the vending machine sells its own products.

The secondary reference, Chao, discloses payment adjustment rules for determining salesmen's compensation (see, e.g. Chao paragraph [0071]). Chao also discloses general concepts regarding object oriented programming such as subclasses inheriting from parent classes (Chao paragraph [0025]). Chao does not, however, disclose, teach, or suggest combining a first allocation rule of a node with a second allocation rule inherited from a parent node as claimed. Chao's inheritance is of object oriented class definitions, not allocation rules. The other reference, Chang, directed to the assignment of sales credit in an organization hierarchy, also fails to cure the deficiencies of Walker.

Because Walker, Chao, and Chang do not disclose, teach, or suggest the above limitations, no combination of the references can render the claims unpatentable under § 103. Therefore, Applicants respectfully request withdrawal of the rejections of the claims and all claims depending therefrom.

As acknowledged by the Examiner, Walker does not disclose an allocation manager configured to make one or more attempts to allocate a member of the plurality of transactions among the plurality of business objects.

Chao paragraph [0072] is cited for this limitation with the proposed motivation that combining this with Chao would have been to define and restrict user access and usage. Applicants respectfully disagree that this limitation is taught by Chao. The "user manager application 140" discussed in Chao paragraph [0072] defines and restricts user access and usage of Chao's application suite. However, defining and restricting user access does not amount to making attempts to allocate a transaction, such as a sales commission, among a plurality of business objects, such as salesmen. Chao's user manager application is for managing *users* and

Appl. No. 10/691,941
Amdt. dated November 3, 2009
Reply to Office Action of August 4, 2009

PATENT

is not “an allocation manager configured to make one or more attempts to allocate a member of the plurality of transactions among a plurality of business objects” as claimed.

For at least the above reasons, Applicants respectfully request withdrawal of the rejections of the claims and all claims depending therefrom.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 925-472-5000.

Respectfully submitted,

A handwritten signature in black ink, appearing to be "Mark Mathison", is written over a dotted line. The signature is somewhat stylized and includes several diagonal strokes and loops.

Mark Mathison
Reg. No. 57,556

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 925-472-5000
Fax: 415-576-0300
MPM:
62291677 v1